IN THE CLAIMS:

The text of all pending claims, (including withdrawn claims) is set forth below. Cancelled and not entered claims are indicated with claim number and status only. The claims as listed below show added text with underlining and deleted text with ~~strikethrough~~. The status of each claim is indicated with one of (original), (currently amended), (cancelled), (withdrawn), (new), (previously presented), or (not entered).

Please AMEND claims 1, 3, 7-10, 12-18 and ADD new claims 19 - 22 in accordance with the following:

1. **(currently amended)** A checksum generation apparatus, comprising:

a control unit which, in response to information on a predetermined length, outputs a control signal when an amount of data corresponding to the predetermined length is received;

an addition unit which ~~receives data, performs~~performs an addition on the received data, and, in response to the control signal, outputs an addition result; and

a conversion unit which converts the addition result to a checksum.

2. **(original)** The checksum generation apparatus according to claim 1, wherein the checksum has a value of 16 bits.

3. **(currently amended)** The checksum generation apparatus according to claim 1, wherein the addition unit receives data in units of ~~32 bits plus (an integer X 16 bits)~~ n*16 bits, where n is an integer greater than or equal to 2, and performs an addition on the received data.

4. **(original)** The checksum generation apparatus according to claim 1, wherein the conversion unit divides the addition result into a sum and a carry, partitions the sum into 16-bit segments, and adds the 16-bit segments to the carry, thereby obtaining a final sum.

5. **(original)** The checksum generation apparatus according to claim 4, wherein, when a carry occurs in the final sum, the conversion unit excludes the carry from the final sum and adds the carry to the carry-excluded final sum, thereby outputting a carry-added final sum.

6. **(original)** The checksum generation apparatus according to claim 5, wherein the conversion unit outputs a 1's complement value of the carry-added final sum as a 16-bit checksum.

7. **(currently amended)** The checksum generation apparatus according to claim 1, wherein the addition unit comprises:

an adder ~~for~~ adding the received data in units of ~~32 bits plus (an integer X 16 bits)~~ n*16 bits, where n is an integer greater than or equal to 2; and

a carry adder ~~for~~ adding carries generated in the adder.

8. **(currently amended)** The checksum generation apparatus according to claim 1, wherein the conversion unit comprises:

a partial sum addition unit ~~for~~ excluding a carry from the addition result, partitioning ~~a~~ the carry-excluded addition result into 16-bit segments, adding the 16-bit segments, ~~thereby obtaining~~ to obtain a partial sum;

a first adder ~~for~~ adding the carry to the partial sum;

a second adder ~~for~~ adding an addition result of the first adder and a carry occurring in the addition result of the first adder; and

a complement calculator ~~for~~ outputting a 1's complement value of the addition result of the second adder.

9. **(currently amended)** A method of generating a checksum, the method comprising ~~the steps of~~:

~~(a)~~ adding input data until a predetermined control signal is received;

~~(b)~~ outputting a sum and a carry obtained from the addition result when the control signal is received; and

~~(c)~~ adding the sum and the carry and converting the ~~addition result~~added sum and carry to a checksum.

10. **(currently amended)** The method of generating a checksum according to claim 9, wherein the checksum ~~has~~ is converted to a value of 16 bits.

11. **(original)** The method of generating a checksum according to claim 9, wherein the control signal is output when an amount of data corresponding to input data length information is received.

12. **(currently amended)** The method of generating a checksum according to claim

9, ~~wherein, in the step (a), data~~wherein the data is received in units of n*16 bits, where n is an integer greater than or equal to 2 ~~32 bits plus (an integer X 16 bits)~~ and an addition is performed on the received data.

13. (**currently amended**) The method of generating a checksum according to claim 9, wherein, ~~in the step (c), the~~the adding of the sum and the carry comprises:

partitioning ~~addition result is divided into a sum and a carry,~~ the sum ~~is partitioned~~ into 16-bit segments, and

adding the 16-bit segments ~~are added~~ to the carry, ~~thereby~~ to obtain a final sum~~being obtained~~.

14. (**currently amended**) The method of generating a checksum according to claim 13, wherein, when a second carry occurs in the final sum, ~~in the step (c), the~~the second carry is excluded from the final sum and added to the carry-excluded final sum, ~~thereby~~ to output a carry-added final sum~~being output~~.

15. (**currently amended**) The method of generating a checksum according to claim 14, ~~wherein, in the step (c),~~ further comprising:

outputting a 1's complement value of the carry-added final sum ~~is output~~ as a 16-bit checksum.

16. (**currently amended**) The method of generating a checksum according to claim 9, wherein the ~~step (b) further comprises the steps of~~outputting of the sum and carry comprises:

~~(b1)~~ adding the received data in units of n*16 bits, where n is an integer greater than or equal to 2~~32 bits plus (an integer X 16 bits)~~; and

~~(b2)~~ adding carries generated in ~~the step (b1)~~adding of the received data.

17. (**currently amended**) The method of generating a checksum according to claim 9, wherein the adding of the sum and the carry and the converting of the added sum and carry to a checksum comprises: ~~the step (c) furthe r comprises the steps of:~~

~~(c1)~~ excluding ~~a~~ the carry from the addition result, partitioning a carry-excluded addition result into 16-bit segments, and adding the 16-bit segments, ~~thereby obtaining~~to obtain a partial sum;

~~(c2)~~ adding the carry to the partial sum to obtain a second addition result and a second carry;

~~(c3)~~ adding ~~an addition result of the step (c2)~~the second addition result and ~~a~~ the second carry ~~occurring in the addition result~~to obtain a third addition result; and

~~(c4)~~ outputting a 1's complement value of the third addition result as the checksum ~~of the step (c3)~~.

18. **(currently amended)** A computer-readable storage medium ~~where~~ storing a program ~~executed~~ executable by a computer ~~is stored~~, the program comprising instructions for enabling the computer to perform a method of generating a checksum, wherein the ~~method~~ program comprises: ~~the steps of:~~

~~a)~~ instructions for adding input data until a predetermined control signal is received;

~~(b)~~ instructions for outputting a sum and a carry obtained from ~~the~~ an addition result when the control signal is received; and

~~(c)~~ instructions for adding the sum and the carry and converting the ~~addition result~~added sum and carry to a checksum.

19. **(new)** An apparatus for generating a checksum, the apparatus comprising:

a first adder adding a predetermined number of bits of input data received in units of n*m bits, where n is a first integer greater than or equal to 2 and m is a second integer, to obtain a first sum having n*m bits and a first carry;

a partial sum calculator partitioning the first sum into n segments of m bits and adding the m-bit segments to obtain a second sum;

a second adder adding the second sum and the first carry to obtain a third sum and a second carry;

a third adder adding the third sum and the second carry to obtain a fourth sum; and

a complement calculator outputting a 1's complement value of the fourth sum as an m-bit checksum.

20. **(new)** An apparatus for generating a checksum, the apparatus comprising:

a first adder adding a predetermined number of bits of input data received in units of n*m bits, where n is a first integer greater than or equal to 2 and m is a second integer, to obtain a first sum having n*m bits and a first carry;

a partial sum calculator partitioning the first sum into n segments of m bits and adding the m-bit segments to obtain a second sum;

a second adder adding the second sum and the first carry to obtain a third sum; and

a complement calculator outputting a 1's complement value of the third sum as an m-bit checksum.

21. **(new)** A method of generating a checksum for input data received in units of n*m bits, where n is a first integer greater than or equal to 2 and m is a second integer n, the method comprising:

adding a predetermined number of the input data units without dividing the input data units into smaller units, to obtain a first sum having n*m bits and a first carry;

partitioning the first sum into n segments of m bits and adding the m-bit segments to obtain a second sum;

adding the second sum and the first carry to obtain a third sum; and

outputting a 1's complement value of the third sum as an m-bit checksum.

22. **(new)** The method of claim 21, wherein:

if a second carry is generated in the adding of the second sum and the first carry, the method further comprises:

adding the third sum and the second carry to obtain a fourth sum, and

replacing the third sum with the fourth sum in the outputting of the 1's complement value as the m-bit checksum.